# CERTIFICATION OF SDRS IN NEW PUBLIC AND GOVERNMENTAL SECURITY SYSTEMS

Stefan Nagel[*], Volker Blaschke[*], Jens Elsner[*], Friedrich K. Jondral[*],
Dimitrios Symeonidis[+]
[*)] Institut für Nachrichtentechnik, Universität Karlsruhe (TH), Germany
{nagel;blaschke;elsner;fj}@int.uni-karlsruhe.de
[+)] JRC Joint Research Centre – European Commission, Ispra, Italy
dimitrios.symeonidis@jrc.it

## ABSTRACT

The today's networks of Public and Governmental Security (P&GS) systems are characterized by a heterogeneous structure. Since the last years the demand for interoperability between such networks has been increased, significantly. Furthermore, flexible hardware platforms and portable waveform applications are required in order to provide flexible and adaptable solutions to costumers. In order to ensure interoperability between terminals of different vendors and compatibility of waveforms leads to a certification process which each system has to pass. Both aspects, platform as well as waveform certification needs will be discussed in the following paper.

## 1. INTRODUCTION

In this paper the technical requirements for certification of SDR platforms and its components are described. The platform-waveform paradigm followed by major SDR Architectures (JTRS SCA, OMG SWRadio) is adhered to and necessary tests to assure seamless interoperability and to facilitate waveform portability are identified. The study concerns the certification of both handsets and base stations. The connection between certification and standardization is emphasized and the benefits of a reference implementation are discussed. Apart from API availability and correct implementation testing for the platform and the waveform, the testing of non-functional requirements such as performance or compliance to national transmission regulations is suggested.

Two terms need to be defined before taking on the issue of SDR certification.

*Portability* is a term with multiple meanings, depending on the context. Two major interpretations are: run-time portability and build-time portability. Run-time portability is an idealistic goal, where the actual binary can be exchanged between platforms and retain full functionality when executed, without any modifications. In contrast, build-time portability is the porting of a component's code to a new waveform with a limited amount of modifications. This view on portability is more appropriate in a realistic scenario. In the following, when portability is mentioned, it refers to build-time portability.

Secondly, the term *interoperability* is often used in publications with varying meaning. Here, interoperability refers to interoperability between radio systems. This may include interoperability:

- between two SDR sets comprised of the same platform and the same waveform,
- between two SDR sets comprised of different platforms running the same (ported) waveform,
- between two SDR sets comprised of different platforms and different waveforms, which are implementing the same radio standard, or
- between an SDR set and one or more legacy radios.

After clarifying these two terms, the certification process in SDR systems can be described in detail.

## 2. CERTIFICATION PROCESS IN SDR SYSTEMS

Certification of SDR platforms and waveforms is necessary to facilitate portability of waveforms between platforms. Certification verifies that interfaces closely follow the referenced architectural framework. This in turn avoids duplication of effort and increases competition by allowing SDR components for platforms or waveforms to be developed independently. Minimum porting effort encourages innovation and lowers the market entry barrier.

Developing a certification process is per se a challenging goal; developing an SDR certification process is even more challenging because Software Defined Radio is a technology that involves two very different disciplines:

software and radio. Each of these disciplines is wide, mature and well studied; however the cross-section of both, which is Software Defined Radio, is a new technology that is not yet as mature as each one of them individually.

## 3. PLATFORM CERTIFICATION

The certification of an SDR platform includes, first of all, the verification of the APIs each component advertises to other platform components or to waveform application components. This verification includes not only their existence, but also their correct behavior. It is achieved by a custom-built test application that is generated by the certification software suite. The certification suite takes as input the underlying platform as well as the component under test and develops a test application that is to be deployed, loaded and executed on the target platform. This test application then performs a series of compliance, performance and stress tests on the platform component and returns a test report with the results of each subtest. The test application is similar to a waveform in terms of structure and deployment mechanism; however, such a test waveform would probably not transmit or receive anything, with the possible exception of the testing of the RF frontend components. The benefits of following the test application approach for SDR platform certification are that a uniform set of rules and deliverables can be defined for all necessary tests, and that no extra, specialized features need to be provided by the platform for testing purposes; the test waveform is loaded and executed like any other waveform.

ESRA is the Architectural Framework for European Software Defined Radio (SDR), defined inside the WINTSEC project.

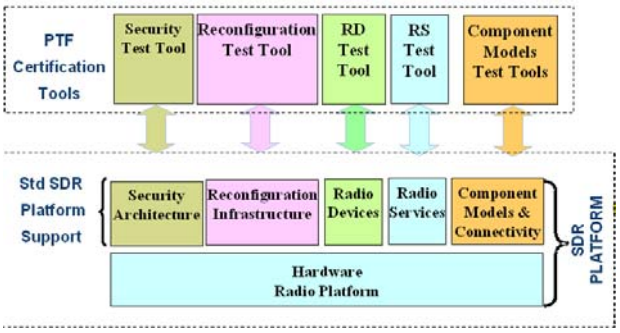


**Figure 1: Paradigm for ESRA Components Certification**

The ESRA Framework defines seven main categories of platform components: the Reconfiguration Architecture, the GPP component models, the DSP component models, the FPGA component models, the Connectivity mechanisms between these component models, the Radio Domain Devices and the Radio Domain Services.

### 3. 1 Reconfiguration Infrastructure

The Reconfiguration Infrastructure uses the meta-data related to a waveform to deploy the waveform's components, interconnect them, and configure the underlying platform resources to fulfill the needs of the Waveform Functionality. Certification of the Reconfiguration Infrastructure will include verifying that each component of the test waveform is deployed to an execution unit of the correct type and of sufficient resources to satisfy the real-time processing requirements of the component, and with sufficiently fast connections to the test waveform's other components. At the same time, the certification test will make sure that all platform subsystems (Devices and Services) have been correctly reconfigured to satisfy the test waveform's requirements, and that an acknowledgment of successful instantiation (or of failure) of the test waveform is sent to the operator. Two additional issues for certification are the correct removal of a waveform by the Reconfiguration Infrastructure (the right order of the removal of components and the release of platform resources), while other waveforms running on the platform must stay intact, as well as the speed of these operations, which is especially important for Cognitive Radio (CR) functionality. Finally, it is important to make sure that if the deployment of one component fails, the rest of the components are cleanly released, the platform subsystems return to their previous state, and a clear message is sent to the operator, with detailed information appended to the system log.

### 3.2 Component Models & Connectivity Mechanisms

A Component Model is a complete set of software engineering assumptions attached to an Execution Unit (a GPP/DSP/FPGA processing core), which allows a Waveform to realize its components on the Execution Unit. These assumptions include a set of Technical Services (e.g. scheduling, memory management, logging subsystem and memory, internal connectivity), a standardized interface to the Reconfiguration Infrastructure, and a standardized way to share Execution Resources. It is important to check that each Execution Unit can host functional components of different waveforms, as well as the availability, predictable and compliant behavior, and the ro-

bustness of the related Technical Services, the Reconfiguration Infrastructure interface and Execution Resources sharing.

While Inner Connectivity, i.e., between components deployed on the same Execution Unit, is part of the Technical Services provided by a Component Model, Cross Connectivity Mechanisms are the techniques that allow components deployed on different Execution Units to communicate and share resources, as well as the connectivity between Execution Units and Radio Domain Devices & Services. After verifying the availability of the correct API, the call-and-response procedure of requesting resources should be checked, including the connection speed, data format and priority handling.

### 3.3 Radio Domain Devices & Services

Radio Domain Devices provide an abstraction layer for all the hardware components that a waveform might need, which are not a GPP, a DSP or an FPGA. These include one or multiple RF frontends, IO devices (audio, HCI, data, etc) and other support devices (e.g. a GPS or a frequency reference). Radio Domain Services are software-only artifacts of the SDR platform that provide Waveform Support (e.g. Vocoder, Internet Protocol, or Retransmission) or Platform Management (e.g. Configuration, Fault Management, etc). Their certification would include a standard way to advertise their existence, the correct behavior of the related API in normal mode, as well as their correct response and stability in case of erroneous input.

### 3.4 Security Architecture

It needs to be clarified that what is being certified here is not the security itself (Confidentiality, Integrity, Authentication and Availability) of the over-the-air communication protocol; this would be part of the interoperability certification, as the above features are part of each wireless standard. Instead, what is intended by the term Security Architecture is a set of interfaces between Software Applications and the Security Functions, such as Security Devices or Services. Certifying the availability and compliant operation of these interfaces would permit the sharing and reuse of Security Components between Waveforms, thus increasing the overall quality of Information Assurance in SDR Sets and avoiding the duplication of efforts.

### 3.5 Platform-wide certification

The above certification tests targeted each specific component of an SDR platform, with the goal of verifying that the component under test complied with the related guidelines in the ESRA architectural framework. However, to achieve the goals of ESRA certification (wireless interoperability and waveform portability), it is important to execute some additional tests, which target the entire platform and verify the correct integration of the individual components. Successful integration assures that all the components are installed, instantiated and connected to each other in the intended collaborative way, for example that a software component deployed on an execution unit can communicate (through the connectivity mechanisms) to an installed Radio Device or access a running Radio Service. Furthermore, such tests would make sure that the SDR platform can actually perform radio transmissions and receptions, since an SDR set is not only a software platform, but also a radio platform.

### 3.6 Performance

Another important parameter to be tested and certified is the performance of a platform to execute the tasks required in a dependable, reliable and timely manner. The speed and reliability of the reconfiguration procedure (changing waveform parameters on-the-fly, loading and unloading waveforms) is important in the SDR world and even more so in the CR world. Minimum end-to-end reconfiguration times can be defined via a usage scenario: it needs to be proved that the SDR platform can dependably reconfigure itself in a given amount of time, for a target configuration of a given complexity, but independently of the starting platform configuration. The reliability and robustness of the SDR platform can be examined through stress-testing, through the attempted loading of erroneous waveforms, or through the reception of malformed signals. Furthermore, it is important to test that the SDR platform correctly handles events coming from its sensors, such as low battery status or thermal warnings, independent of the platform's configuration.

Besides, the metrics used by platform components to advertise their performance capabilities need to be matched with the performance requirements for deployment of each waveform component. The maximum values of these metrics (i.e. when not running any waveforms) need to exceed the waveform's performance requirements, before any effort of porting the given waveform to the platform is undertaken. Moreover, the instantaneous values of these metrics need to be compared by the Reconfiguration Infrastructure with the minimum performance requirements of an SDR waveform before loading and executing it, especially in the case of multi-channel operation.

### 3.7 Certification tools

As for the tools that can be used to perform these tests, from a preliminary study it became clear that in some cases, existing tools can be adapted for the task at hand, while in other cases, new tools will need to be developed, mainly due to the fact that ESRA introduces several new concepts to the Software Defined Radio world, like DSP & FPGA Component Models, or Radio Devices & Services.

## 4. WAVEFORM CERTIFICATION

While the SDR platform provides the environment and necessary application services for a specific waveform, the waveform application itself implements the core functionality of the radio system. Due to this, waveform certification is paramount to the whole radio certification process. In the context of Certification of SDRs in public and governmental security systems Waveform certification consists of three parts:

- interoperability certification,
- portability certification,
- referenced standard certification.

### 4.1 Certification for interoperability

Certification for interoperability is a well studied subject. For any given wireless standard, established procedures and tools already exist. These interoperability tests usually consist of extensive test trials, which include verifying adherence to, e.g., spectrum mask, transmission power, delay and timing. Interoperability requirements are, in the majority of cases, defined in the corresponding air interface standards. If an SDR-set is standard compliant and hence interoperable with a legacy device, it's also interoperable with any other standard compliant SDR-set implementing the same air interface (but not necessarily running the same waveform software). Due to this, a single SDR reference platform could be used to do the interoperability tests, after it was tested against the legacy devices as shown in Figure 2.

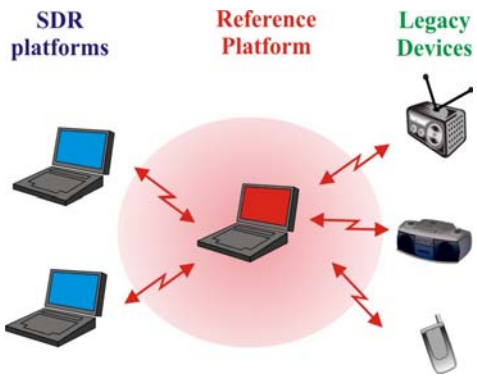


**Figure 2 Certification for interoperability**

### 4.2 Certification for portability

Portability is a measure for the costs of porting a waveform from one platform to another. Run-time portability means that no costs occur by porting because one can execute the binaries on the target platform. Unfortunately, this is utopian, due to the fact that, in the majority of cases, code for a specific hardware architecture does not run without modifications on any other hardware architecture. While certifying build-time portability another problem occurs: how to measure portability? Different to interoperability, there is no clear-cut way to say a waveform is portable or not. Portability does also depend on the underlying platforms. While the port between two similar platforms is relatively easy, another port between two completely different hardware architectures can be very cost intensive. Overall, it is inexpedient to certify the waveform itself for portability. A different approach is more promising: certification of the waveform development process. The development of portable waveforms follows the Model Driven Architecture [1], an initiative by the OMG which introduces four models on the way to the finished waveform:

1. The Computation Independent Model (CIM) as a functional description.
2. The Platform Independent Model (PIM), which implements the functionality without platform aspects.
3. The Platform Specific Model (PSM) as the extension of the PIM with platform aspects.
4. The Code, that is executable on a specific hardware.

While the probability of the reuse of the PSM and the Code is very low for different platform architectures, the CIM and the PIM are very good inputs for a waveform developer. According to [2] the documentation of code and the easy understanding of algorithms are essential for a portable waveform. The understanding of the algorithms is supported by the functional description and the documentation of code follows coding standards. An approach of implementing a waveform after the MDA can be found in [3].

### 4.3 Certification for ESRA

In the certification process for ESRA, compliance has to be verified in four areas:

1. Compliance with the component models and the connectivity mechanism.
2. Compliance with Radio Domain Devices.
3. Compliance with Radio Domain Services.
4. Compliance with the Reconfiguration Infrastructure.

The proposed processes are illustrated in Figure 3.

*Compliance with the component models and the connectivity mechanism*

In the context of ESRA, Component models are defined as a complete set of software engineering assumptions that define a formal way to realize software modules (the Components) of a Waveform Implementation. This assures a high degree of Platform Reconfigurability and Waveform Portability. These components have been verified for:

- Functionality
- Timing

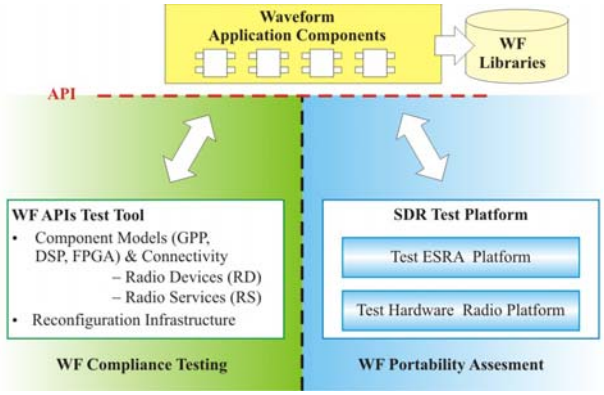


**Figure 3: Waveform Certification Requirements**

- Memory
- Concurrent behavior
- Physical interaction
- Precise physical interaction in SDRs with amplifiers or synthesizers

The first four are well-known in the world of software components while the last dimension has a special role in physical systems like a Software Defined Radio. This could be the configuration of a synthesizer or the DDC (Digital Down-Converter) in an FPGA. It is necessary to verify that the output of the software really has the intended effect on the physical world. This makes software verification much more difficult than simply verifying that the software properly produces a set of discrete output values.

Furthermore the components have to be encapsulated in such a way that it is impossible to identify whether the connection to another component is just logical (to another component on the same model or on another model) or physical (to another component on another execution unit). This has to be done by a middleware such as CORBA or through a predefined set of instructions such as POSIX or APIs.

*Compliance with Radio Domain Devices*

As described above, the Radio Domain Devices provide the waveform with access to Radio Platform Subsystem functionalities like I/O, Support and Radio Frequency Devices. Due to the fact that I/O and Support devices are independent from the waveform, the compliance can be verified by interface testing, especially by certifying the APIs of these devices.

A Radio Frequency Device defines an API for transmission and reception of I/Q base band samples. A standardized and widely accepted API is not yet available; the *SDR Forum Task Group on Transceiver Subsystem Interfaces* is currently working towards this goal. To maximize waveform portability, waveforms should use a standard API and, if possible, avoid vendor specific extensions. The API should define a base interface and extensions, which cover the most common use-cases (e.g. frequency hopping support).

As a side note, an API that separates signal processing from the transceiver allows *I/Q base band certification for interoperability* of a waveform. Transceiver modules can then be developed and certified separately, which makes "all software" certification for waveforms possible. Waveform I/Q baseband test cases could then be defined as part of the PIM, assuring compliance with the wireless standard implemented.

*Compliance with Radio Domain Services*

**Table 1: Summary of waveform certification**

| Testing | Aspect | Requirement |
|---|---|---|
| API | Specification coverage | Provide assurance that the implementation meets the minimum API coverage to qualify for particular compliance level. |
| | Exception integrity | Provide assurance that erroneous usage raises the specified Exception(s). |
| Consistency | XML-model | Component model artifacts (e.g. XML) shall be tested for referential integrity, consistency with source model and standards compliance. |
| | Decomposition | Providing that a consistent decomposition is applied to a particular waveform, testing can be applied at the component level to provide assurances that the waveform can interoperate. |
| Operational | Component re-use | Where a component implementation is based on a foreign infrastructure, an assurance shall be given that such a component implementation can be re-used. For example in the case of CORBA infrastructure (ORB library), this measure would reflect the degree of portability across ORBs afforded by the component implementation. |

The Radio Domain Services provide Waveform Support Services and Platform Management Services. While the Management Services are non-functional services which are not used by the waveform, the access to Support Services, e.g. to the Vocoder, is essential. Therefore, the interface has to be tested in a way that the correct APIs are accessed in a correct manner.

*Compliance with the Reconfiguration Infrastructure*

The Reconfiguration Infrastructure is provided with the description of the waveform in meta-files (e.g. written in XML) and deploys the different components on the available execution units. These meta-files accompanying each Waveform Application need to be tested and certified in the following areas:

- the format and all the elements of the meta-files need to be in a defined format, structure and terminology,
- the type of Execution Unit a Waveform Module "wants" to be deployed according to the meta-files should match the executable binary format of the Waveform Module,
- the minimum performance of the Execution Unit (processing speed, cache memory, connectivity bandwidth) required by each Waveform Module should be declared in the meta-files,
- the meta-files should declare which Radio Domain Devices and Services the Waveform Application and its Functional Modules need to make use of, so the Reconfiguration Infrastructure can check for their existence, availability and capabilities.

Table 1 summarises the waveform certification testing requirements implicitly defined in ESRA. It is expected that a system, which meets these requirements and applies them successfully to a particular waveform implementation will provide a best-effort level of assurance that the waveform can be ported across certified platforms.

## 5. CONCLUSION

Certifying an ESRA-compliant Software Defined Radio has the dual goal of Waveform Portability and Wireless Interoperability. Platform Certification is achieved through the execution of a tailored test waveform, which tests each platform component individually, then tests whole platform integration, and finally tests platform performance and reliability. Waveform Certification requirements are divided into interoperability requirements, portability requirements and ESRA Framework-related requirements.

## ACKNOWLEDGEMENT

This work has been performed in the framework of the EU funded project WINTSEC. The authors would like to acknowledge the parts of this paper coming from contributions of their colleagues from WINTSEC consortium.

## 6. REFERENCES

[1] Model Driven Architecture (MDA), Architecture Board ORMSC, Object Management Group, July 2001.

[2] C. Epifanio, M. Uhm, "The Myth of Code Portability", SDR Forum Technical Conference 2007

[3] S.Nagel, D. Epple, F. Jondral, "Implementing the TETRA physical layer on Lyrtech's SFF SDR Development Platform", SDR Forum Technical Conference 2008